

USER'S MANUAL
990-331
REVISION B: January 2002



MODEL HF25
ADVANCED SERIAL DATACOM
COMMUNICATIONS INTERFACE
DOS VERSION

Copyright © 1997, 2002 Unitek Miyachi Corporation

The engineering designs, drawings and data contained herein are the proprietary work of UNITEK MIYACHI CORPORATION and may not be reproduced, copied, exhibited or otherwise used without the written authorization of UNITEK MIYACHI CORPORATION.

Unitek Miyachi
1820 South Myrtle Avenue
P.O. Box 5033
Monrovia, CA 91017-7133
Telephone: (626) 303-5676
FAX: (626) 358-8048
e-mail: info@unitekmiyachi.com

Printed in the United States of America.

REVISION RECORD

Revision	EO	Date	Basis of Revision
A	--	2/97	Original Release.
B	19146	01/02	Updated data and format.

Table of Contents

Page

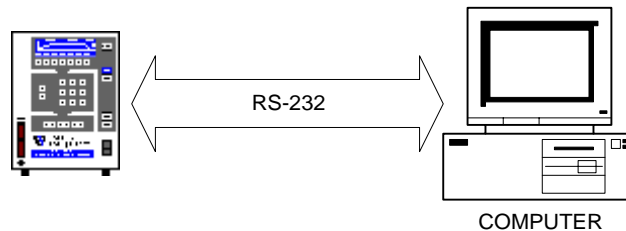
Section 1. General Description	1
HF25 Serial Communication	1
HF25 Communication Options	1
HF25 Datacom Kit (10-347-01)	2
RS-232 Data Communication Operation	2
Purpose	2
Making RS-232 Connections	3
RS-232 Serial Connector Information	4
RS-485 Data Communication Operation	5
Purpose	5
Making RS-485 Connections	5
RS-485 Cabling Information	6
RS-485 To RS-232 Interface Converter Information	7
Section 2. Remote Data Collection	8
Weld Report	8
Example of a Weld Report	8
Simple Data Collection	9
Examples of Remote Data Collection Commands	9
Section 3. Remote Programming	10
Command Summary	10
Examples of Remote Programming Commands	11
Appendix A. Weld Status Messages	A-1
Appendix B. Host Originated Commands	B-1
Appendix C. HF25 Originated Commands	C-1

Section 1. General Description

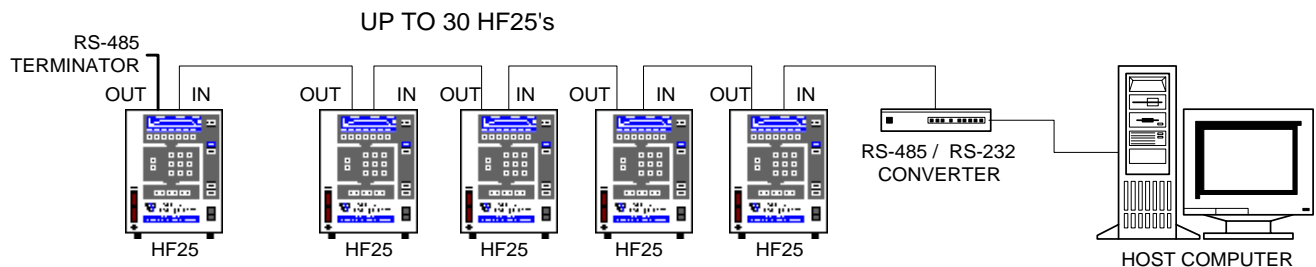
HF25 Serial Communication

The purpose of the HF25 Control serial port is to enable remote data collection and remote programming via a serial communications line between the HF25 Control(s) and a serial data collecting device or host computer. The HF25 Control has two ways of reporting weld information or programming remotely - using RS-232 and RS-485. The RS-232 communication provides a simple way of reporting weld information from a HF25 Control to a computer. To use the RS-232 communication, refer to RS-232 Data Communication Operation. The RS-485 communication provides a way of collect weld reports from one or more HF25 Controls using one host computer. To use the RS-485 communication, refer to RS-485 Data Communication Operation. There is no difference between the operations of RS-232 communication and RS-485 communication other than that RS-485 can be daisy-chained and RS-232 can not.

RS-232 COMMUNICATION



RS-485 COMMUNICATION



HF25 Communication Options

To establish communication through the Serial Port connector, the Communication options should be set to match the communication configuration of the host computer. The communication options are set from the HF25 COMMUNICATION menu.

OPERATING INSTRUCTIONS

- Baud Rates:** The baud rate of the HF25 Weld Control can be configured at 1200, 2400, 4800, 9600, 14.4K, 19.2K, or 28.8K. The default setting is 9600.
- Parity:** The HF25 always communicates in 8 bits with no parity and 1 stop bit.
- Datacom Role:** The HF25 can be configured as a MASTER or SLAVE. When “MASTER” is selected for the DATACOM ROLE, the HF25 sends out the weld report via the Serial Port after each weld is made. When “SLAVE” is selected for the DATACOM ROLE, the HF25 sends out the weld report only if it is requested by the host computer. The default setting is SLAVE.
- I.D. Number:** To identify each HF25 Weld Control connected to one RS-485 communication line, the host computer needs to know the identification number of each HF25. The I.D. Number can be any number from 0 to 99. The default setting of I.D. NUMBER is 1.

HF25 Datacom Kit (10-347-01)

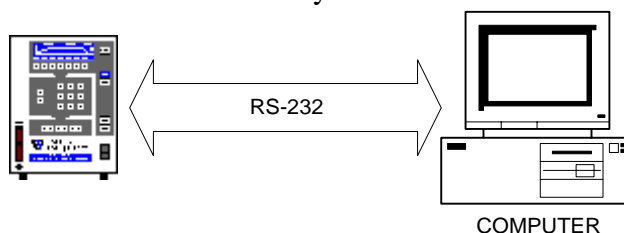
The HF25 DATACOM KIT provides all the tools necessary to collect weld data from the HF25 Control using a host computer. It should be consist of:

1. 9 pin male to 9 pin female cable
2. 9 pin to 25 pin serial adapter.
3. RS-485 terminator.
4. DATACOM manual
5. HF25 data collection software.

RS-232 Data Communication Operation

Purpose

The purpose of the RS-232 port is to enable remote data collection and remote programming via a RS-232 serial communication line from one HF25 Control to a serial data collecting device or host computer. This protocol provides a simple way of reporting weld information from a HF25 Control to a computer. Using a RS-232 communication, remote programming is also allowed. However, this section describes simple remote data collection method only.



Making RS-232 Connections

- 1 Turn off the HF25 Control.
- 2 Connect the 9 pin to 9 pin cable to RS-232 connector on the rear panel of the HF25 Control to the RS232 connector on a host computer. Use 9 pin to 25 pin adapter if necessary. The cable and adapter are part of the HF25 DATACOM kit.
- 3 Insert DATACOM diskette to the floppy drive of the computer. The data collection software included in HF25 DATACOM kit provides a simple way of collecting weld reports from a HF25.
- 4 Type INSTALL to install the data collection software. Follow the screen instructions.
- 5 After the installation is complete, type DATACOM from DOS prompt.
- 6 Select SETUP and set COM port to serial port to which the serial cable is connected. Set baud rate to 9600 baud. Set HF25 Communication Role to "SLAVE".

```

< COLLECTING HF25 WELD DATA >
      < COMMUNICATION SETUP >
      BAUD RATE      : 9600
      HF25 I.D.      : 01
      COMMUNICATION PORT : COM1
      FILENAME       : TEST.TXT
      HF25 DATACOM ROLE : SLAVE

```

▲▼ Select, ◀ ▶ Change, or press [ESC] to return.

- 7 Turn on the HF25 Weld Head Control.

OPERATING INSTRUCTIONS

- Set the baud rate of the HF25 Control to 9600. The HF25 serial communication baud rate setting is listed under the HF25 Control COMMUNICATIONS menu.

```

< COMMUNICATION >
1.COMMUNICATION ROLE      : SLAVE
2.I.D. NUMBER            : 01
3.BAUD RATE              : 9600

NUMBER Select, MENU Previous menu

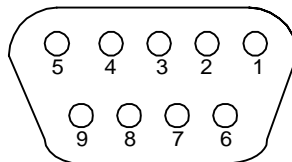
```

- Set the ROLE option in the HF25Weld Head Control COMMUNICATIONS menu to "SLAVE". I.D. NUMBER does not matter at this time.

RS-232 Serial Connector Information

The serial port pin assignment is as follows:

Pin	Name	Description (DCE)
1		
2	TXD	RS-232 Transmit Data
3	RXD	RS-232 Receive Data
4	DSR	Data Set Ready
5	SGND	Signal Ground
6	DTR	Data Terminal Ready
7	CTS	Clear to Send
8	RTS	Request to Send
9	RI	Ring Indicator

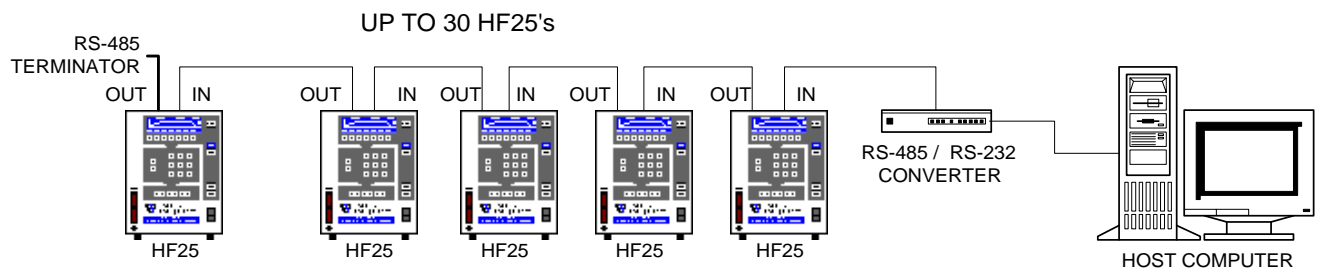


RS-232 Connector on the back panel of the HF25.

RS-485 Data Communication Operation

Purpose

The purpose of the RS-485 ports are to enable remote data collection and remote programming via a half-duplex multi-drop RS-485 communications line between HF25s and a serial data collecting device or host computer. The protocol will be implemented using printable ASCII characters to allow ease of protocol translation, comprehension, and debugging in control systems development.



Making RS-485 Connections

- 1 Turn off the HF25 Control(s).
- 2 Connect the serial connector on a host computer to RS-232/RS-485 converter using 9 pin to 9 pin cable provided with HF25 DATACOM kit. Use 9 pin to 25 pin adapter if necessary.
- 3 Connect the RS-485 IN connector on the rear panel of each HF25 Weld Head Control to the RS-485 connector on the RS-485 to RS-232 converter. There is no standard cable for RS-485; therefore, special cabling is required. Refer to the cabling information in RS-485 CABLING INFORMATION section. If the host computer have a RS-485 communication port, there is no need to use the RS-232 to RS-485 converter.
- 4 Daisy-chain the HF25 Controls if more than one HF25 Controls are to be connected. Use standard 9 pin female to 9 pin female serial cables.
- 5 Turn on the HF25 Control(s).
- 6 Set the baud rate of the HF25 Control(s) and the host computer to the same baud rate. The HF25 serial communication baud rate setting is listed under the HF25 Weld Head Control COMMUNICATIONS menu.
- 7 Set the ROLE option in the HF25 Control COMMUNICATIONS menu to “SLAVE”.
- 8 Set the unit ID number in the HF25 Control COMMUNICATIONS menu. Each HF25 unit should have its own unique ID number to be identified by the host computer.

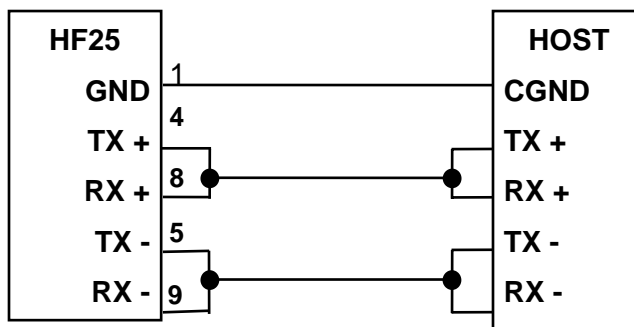
OPERATING INSTRUCTIONS

RS-485 Cabling Information

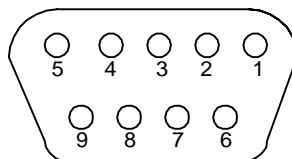
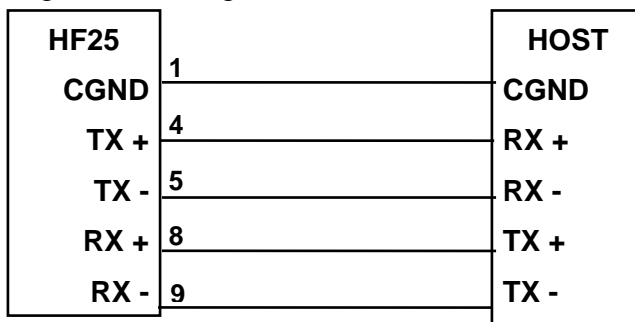
The HF25 Control RS-485 OUT port and RS-485 IN port have following pin assignments for RS-485 communication. There is no difference between RS-485 OUT port and RS-485 IN port.

Pin	Name	Description
1	GND	Signal Ground
4	TXD +	RS485 Transmit Data (+)
5	TXD ~	RS485 Transmit Data (-)
8	RX +	RS485 Receive Data (+)
9	RX ~	RS485 Receive Data (~)

The RS-485 cabling method using 2 wires and a shield:



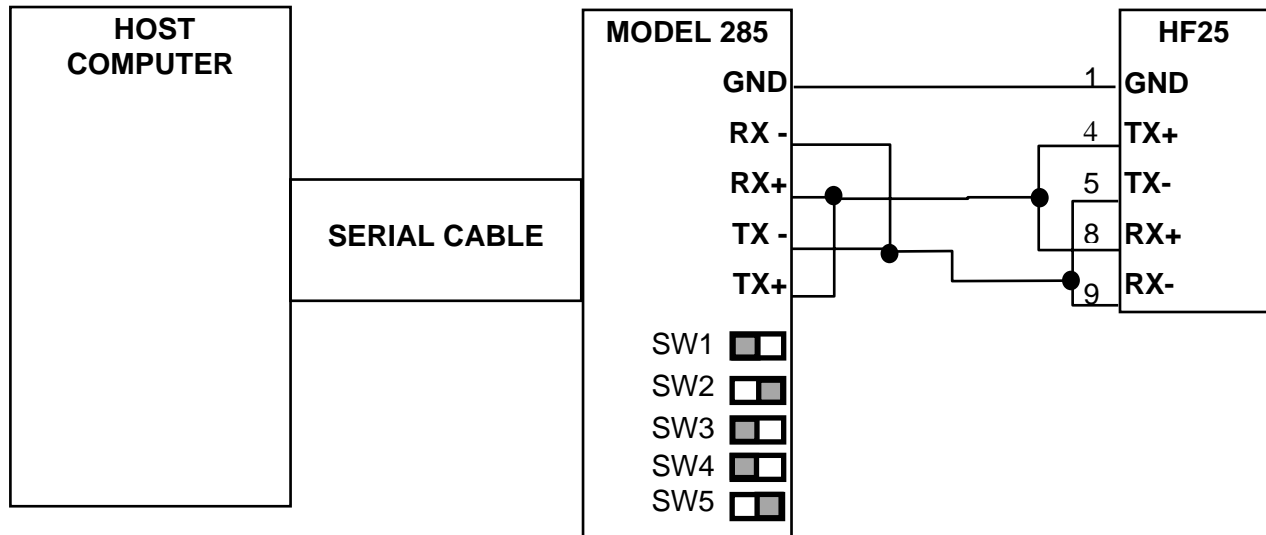
The RS-485 cabling method using 4 wires and a shield:



RS-485 Connector on the back panel of the HF25.

RS-485 to RS-232 Interface Converter Information

Unless the host computer has a RS-485 card, the RS-485 to RS-232 converter is necessary to connect a host computer to the RS-485 port of HF25 Control(s). Use TELEBYTE Model 285 Superverter or equivalent. The followings are the description for how to setup using TELBYTE Model 285 Superverter using 3 wires connection.



Set the jumpers as following:

sw1	ON
sw2	OFF
sw3	ON
sw4	ON
sw5	OFF

OPERATING INSTRUCTIONS

Section 2. Remote Data Collection

The HF25 data communication protocol includes the capability of collecting basic weld information for each individual weld. The HF25 stores weld information for the last 3000 welds.

Weld Report

When *DATACOM ROLE* of the HF25 Control is set to “MASTER”, the HF25 sends out the weld report via the serial ports after each weld is made. When *DATACOM ROLE* of the HF25 Control is set to “SLAVE”, the HF25 sends out the weld report only if it is requested by the host computer. With “SLAVE” setting, the host computer should periodically poll the HF25 to collect the weld data before 3000 welds are reached. Otherwise, data over-run occurs and weld data will be lost starting with the oldest data. The following information is included in the HF25 weld report.

1. The **schedule number** of the weld,
2. The average **current** of 1st weld pulse,
3. The average **voltage** of 1st weld pulse,
4. The **% control** capacity needed to reach the 1st weld pulse,
5. The average **current** of 2nd weld pulse,
6. The average **voltage** of 2nd weld pulse,
7. The % control capacity needed to reach the 2nd weld pulse,
8. The **status** of the weld.

Example of a Weld Report

3,205,217,12,513,452,22,0

The above weld report represents the following information:

1. Schedule number: **3**
2. Average current of the 1st weld pulse: **205 A**
3. Average voltage of the 1st weld pulse: **217 mV**
4. % control capacity of the 1st weld pulse: **12 %**
5. Average current of the 2nd weld pulse: **513 A**
6. Average voltage of the 2nd weld pulse: **452 mV**
7. % control capacity of the 2nd weld pulse: **22 %**
8. Status of the weld: **0 (GOOD)**

For the list of the weld status codes, see *Appendix A, Weld Status Numbers*.

Simple Data Collection

In order to do the simple data collection, the host only needs to send **#ID REPORT NEW *number*** command. **ID** is the identification number of the HF25. ***number*** is a number greater than the number of welds made since the last data collection. Then the HF25 then send all the weld reports since the last data collection and erases all the weld data sent from the weld data buffer. The host should parse the weld report. The weld reports are separated with `<crLf>`. The fields within the report are separated with a comma. See SECTION 3 & 4 for additional remote data collection commands.

Examples of Remote Data Collection Commands

#1 REPORT OLD 10 <crLf><lf>

Host is requesting the HF25 with id #1 to send 10 weld reports from the accumulated weld reports. The weld data counter in the HF25 is decremented by 10. The corresponding HF25 with id #1 will response with:

```
#1 REPORT 10 <crLf>
3,205,217,12,513,452,22,0 <crLf>weld report #1
3,211,216,13,511,451,22,0 <crLf> weld report #2
3,202,212,14,512,453,22,0 <crLf> weld report #3
3,203,217,12,513,456,22,0 <crLf> weld report #4
3,207,218,11,513,457,22,0 <crLf> weld report #5
3,206,222,12,514,458,22,0 <crLf> weld report #6
3,205,214,14,513,461,22,0 <crLf> weld report #7
3,208,212,13,512,439,22,0 <crLf> weld report #8
3,209,218,15,515,451,22,0 <crLf> weld report #9
3,202,218,16,512,455,22,0 <crLf> weld report #10
<lf>
```

#1 REPORT NEW 1 <crLf><lf>

Host is requesting the HF25 with id number 1 to send the latest weld reports. The weld data counter is reset to 0. The corresponding HF25 with id #1 will response with

```
#1 REPORT 1 <crLf>
3,205,217,12,513,452,22,0 <crLf>weld report #1
<lf>
```

Section 3. Remote Programming

The portions of the protocol that include the physical layer (hardware electrical interconnect), the link layer (framing, data encoding, duplex control), the network layer (source-destination identification) and the transport layer (packet framing & token control) are implemented in the HF25. The HF25 must always be in the Slave Mode to send data in a polled request-response synchronous operation utilizing a packetized token-passing-like control & accepting input commands.

The multi-drop signal synchronization control utilizing a token-passing-like algorithm uses the unit identification portion of the command screen as the token, which is the pound sign (#) followed by the unit ID number. When no information is being passed, the host passes an empty token, which is a packet consisting of the token followed by the end of packet sequence (<crLf><lf>). If the HF25 has a message to return, it sends the message along with the token to the host. Otherwise, if the HF25 has no message to return, it returns an empty token. A message consists of any command and its parameters or other data accompanying the command. Each token-message packet must conclude with an end of packet sequence. The HF25 ignores any packet beginning with a unit ID that does not match it's programmed value, up to the point that an idle line is detected. Thus, at least one character time of idle line is required between packets to wake up all HF25 Weld Controls on the communication line in order to recognize any subsequent packet that may be addressed to them.

Command Summary

Packet format:

#ID **KEYWORD** *parameters* <crLf><lf>

unit identification & token *#ID* (*ID* is any number from 0 to 255, must be left-justified or zero-padded to left)

command keywords:**BOLD**

variable to be replaced by literal:*italics*

required parameters:{enclosed in braces} (one required and only one parameter allowed)

optional parameters:[enclosed in brackets] (zero or more allowed)

required/optional parameters: {[enclosed in braces and brackets]} (one or more allowed)

choice of parameters:separated by vertical bar "|" indicates one *OR* another of choices presented.

range of parameters:*low_end* - *high_end* (separated by hyphen)

end of parameter line:<crLf>(carriage return followed by newline)

end of packet:<lf>(new line - must be preceded by the end of parameter line <crLf>)

Each component (token, keyword, and parameters) will and must be separated by one or more non-printing characters (spaces or tabs) except the end of packet <lf> must follow the end of parameter line <crLf> immediately. Non-printing characters immediately preceding the end of parameter line <crLf> are ignored.

Examples Of Remote Programming Commands

#1 LOAD 10 <crLf><lf>

Host is requesting the HF25 that has ID #1 to load schedule #10. The corresponding HF25 with ID #1 will response with

#1 <crLf><lf>
#1 COUNTER TOTAL <crLf><lf>

Host is requesting the HF25 with ID #1 to send the total weld counter. The corresponding HF25 with ID #1 will respond with

#1 COUNTER ##### <crLf><lf> *where ##### is the current total weld counter.*
#5 SCHEDULE <crLf><lf>

Host is requesting the HF25 with ID #5 to send the current schedule number selected. The corresponding HF25 with ID #5 will respond with

#5 SCHEDULE ##<crLf><lf> *where ## is the current schedule number of HF25*
#5 SCHEDULE READ<crLf><lf>

Host is requesting the HF25 with ID #5 to send the current schedule information. The corresponding HF25 with ID #5 will respond with

#5SCHEDULE ##<crLf> *where ## is the current schedule number of HF25*
FEEDBACK1 ##<crLf> *where ## is the feedback of pulse 1*
FEEDBACK2 ## <crLf> *where ## is the feedback of pulse 2*
SQUEEZE ### <crLf> *where ### is the squeeze time in ms*
UP1 ### <crLf> *where ### is the upslope time of pulse 1 in 100 μ s*
WELD1 ### <crLf> *where ### is the weld time of pulse 1 in 100 μ s*
DOWN1 ### <crLf> *where ### is the downslope time of pulse 1 in 100 μ s*
COOL ### <crLf> *where ### is the cool time in 100 μ s*
UP2 ### <crLf> *where ### is the upslope time of pulse 2 in 100 μ s*
WELD2 ### <crLf> *where ### is the weld time of pulse 2 in 100 μ s*
DOWN2 ### <crLf> *where ### is the downslope time of pulse 2 in 100 μ s*
HOLD ### <crLf> *where ### is the hold time in ms*
ENG1 ### <crLf> *where ### is the energy of pulse 1*
ENG2 ### <crLf> *where ### is the energy of pulse 2*
<lf>

APPENDIX A

WELD STATUS MESSAGES

The last field in the report packet represents the status of the weld made. Status numbers and definitions are listed below.

Status #	Status Message
0	GOOD
1	CHECK CONTROL SIGNALS INPUT STATUS
2	CHECK INPUT SWITCH STATUS
3	FIRING SWITCH BEFORE FOOT SWITCH
4	EMERGENCY STOP ON CONTROL SIGNALS INPUT
5	POWER TRANSISTOR OVERHEATED
6	EMERGENCY STOP - OPERATOR ACTIVATED
7	FIRING SWITCH DIDN'T CLOSE IN 10 SECOND
8	WELD TRANSFORMER OVERHEATED
10	VOLTAGE SELECTION PLUG IS MISSING
11	INHIBIT CONTROL SIGNALS ACTIVATED
12	LOW BATTERY
13	NO CURRENT READING
14	NO VOLTAGE READING
15	LOAD RESISTANCE TOO HIGH
16	NO WELD TRANSFORMER DETECTED
17	WELD SWITCH IN NO WELD POSITION
19	CALIBRATION RESET TO DEFAULT
20	LOWER LIMIT GREATER THAN UPPER LIMIT
21	COOL TIME ADDED FOR DIFFERENT FEEDBACK
23	SYSTEM & SCHEDULE RESET TO DEFAULTS
26	SAFE ENERGY LIMIT REACHED
31	UPSLOPE REQUIRED FOR LOWER LIMIT
32	INPUT TOO LARGE
33	INPUT TOO SMALL
34	PRESS RUN BEFORE WELDING
39	ACCESS DENIED! SYSTEM SECURITY ON

APPENDIX A: WELD STATUS NUMBERS

Status #	Status Message
40	ILLEGAL SECURITY CODE ENTERED
47	ACCESS DENIED! SCHEDULE LOCK ON
65	SCHEDULES ARE RESET
66	SYSTEM PARAMETERS ARE RESET
69	WELD TIME TOO SMALL
71	CURRENT GREATER THAN UPPER LIMIT
72	CURRENT LOWER THAN LOWER LIMIT
73	VOLTAGE GREATER THAN UPPER LIMIT
74	VOLTAGE LOWER THAN LOWER LIMIT
75	POWER GREATER THAN UPPER LIMIT
76	POWER LOWER THAN LOWER LIMIT
77	RESISTANCE GREATER THAN UPPER LIMIT
78	RESISTANCE LOWER THAN LOWER LIMIT
80	WELD STOP - LIMIT REACHED

APPENDIX B: HOST ORIGINATED COMMANDS

These are the commands sent by the host computer, via RS-485 or RS-232 to an HF25.

Command: **STATUS** <crLf><lf>

HF25 state: Any

Description: Requests the HF25 to report the status of the weld data buffer. HF25 returns **STATUS** with either “**OK**” or “**OVERRUN.**”

Command: **COUNT** <crLf><lf>

HF25 state: Any

Description: Requests the HF25 to report the number of weld data accumulated since the last data collection. HF25 returns the **COUNT** even if there is no weld data available.

Command: **LOAD** *schedule_number* <crLf><lf>

HF25 state: RUN state

Description: Selects the *schedule_number* as the currently loaded schedule. *schedule_number* **may** be any number from **0** to **127**. There must be a space between **LOAD** and *schedule_number*.

Command: **SCHEDULE** <crLf><lf>

HF25 state: Any

Description: Requests the HF25 to return the currently selected schedule number.

command: **COUNTER** { **TOTAL** | **HIGH** | **LOW** | **GOOD** } <crLf><lf>

HF25 state: Any

Description: Requests the HF25 to return the HF25 weld counter contents.
TOTAL returns the total number of weld counter.
HIGH returns the out of limits high counter.
LOW returns the out of limits low counter.
GOOD returns the within limits counter.

Command: **REPORT** { **OLD** | **NEW** } *number* <crLf><lf>

HF25 state: Any

Description: Requests the HF25 to send the weld report.

OLD: requests to send the *number* of oldest weld reports since the last data collection. The reported weld data will be erased.

NEW: requests to send the *number* of newest weld reports. All the weld data will be erased after reported.

number: the number of weld data to be sent.

APPENDIX B: HOST ORIGINATED COMMANDS

If the *number* is greater than the number of weld data in the buffer, less than the *number* of weld data will be sent. There must be a space between two fields.

APPENDIX B: HOST ORIGINATED COMMANDS

Command:	ERASE <crLf><lF>
HF25 state:	Any
Description:	Requests the HF25 to erase all the weld reports.
Command:	STATE {READ RUN PROGRAM MENU } <crLf><lF>
HF25 State:	Any
Description:	Commands sentry to identify its current state (" READ " keyword, see STATE under <i>HF25 Originated Commands</i> , or go to either RUN state or PROGRAM state.
command:	SECURITY {OFF SCHEDULE SYSTEM CALIBRATION} <crLf><lF>
HF25 State:	Any
description:	Allows control of the Sentry security mode. " OFF " sets all security status HF25 to " OFF ". " SCHEDULE " sets the schedule lock to " ON ." " SYSTEM " sets the system lock to " ON ." " CALIBRATION " sets the calibration lock to " ON ."
command:	COPY {from_schedule_number} {to_schedule_number} <crLf><lF>
HF25 State:	Any state
description:	Allows one schedule to be copied to another schedule number. From_schedule_number and to_schedule_number may be any number from 0 to 127. Copying a schedule to itself has no effect other than to invoke a schedule printout when "PRINT SCHEDULES/PROGRAMS" is enabled.
command:	SCREEN { TEXT ATTRIBUTE } <crLf><lF>
HF25 State:	Any state
description:	Requests the HF25 to send screen dump. If TEXT is specified, the HF25 sends 320 bytes of the text screen dump. If ATTRIBUTE is specified, the HF25 sends 320 bytes of attribute screen dump.

APPENDIX B: HOST ORIGINATED COMMANDS

Command: **KEY** *key_code* <crlf><lf>
HF25 state: Any
Description: Process the *key_code* as if it was pressed from the front panel.
The following are the *key_codes*:

Key	Dec	Hex
SQUEEZE	58	3A
UP 1	59	3B
WELD 1	60	3C
DOWN 1	61	3D
COOL	62	3E
UP 2	63	3F
WELD 2	64	40
DOWN 2	65	41
HOLD	66	42
UP	30	1E
DOWN	31	1F
0	48	30
1	49	31
2	50	32
3	51	33
4	52	34
5	53	35
6	54	36
7	55	37
8	56	38
9	57	39
PERIOD	46	2E
KA	4	04
V	5	05
KW	6	06
RUN	15	0F
MENU	23	17
ENERGY	10	0A

APPENDIX B: HOST ORIGINATED COMMANDS

PEAK	3	03
-------------	----------	-----------

APPENDIX B: HOST ORIGINATED COMMANDS

command:	ALARM { READ CLEAR SET <i>error_number</i> DISPLAY <i>alarm_message_string</i> } <crLf><lF>																																							
HF25 State:	Any																																							
description:	Provides access to the HF25 alarm logic. When used with the " READ " keyword, the current error condition value is returned. See Appendix A. for list of alarm messages. When the " CLEAR " keyword is used, all alarm conditions are canceled. When the " SET " keyword is used, the host may invoke an error identified by <i>error_number</i> . When the " DISPLAY " keyword is used, an error condition can be created with any message desired. The length of the error message must be limited to 40 characters or less. No help message will be available in connection with this created error message.																																							
command:	SCHEDULE { READ SET } <crLf> <i>parameter_name value</i> [<crLf> <i>parameter_name value</i> <crLf> <lF>																																							
HF25 State:	Any except while welding																																							
description:	Provides control over the HF25 schedule parameters. When used with the " READ " keyword, all parameters pertaining to the currently loaded schedule are returned (see SCHEDULE under HF25 ORIGINATED COMMANDS). When the " SET " keyword is used, the host may set (change) the value of one or more of the parameters pertaining to the currently loaded schedule. The following is a list of valid literal substitutions for the <i>parameter_name</i> and <i>value</i> variables:																																							
	<table><tr><td>ENG1</td><td>{ <i>weld_energy</i> }</td><td>energy amount for pulse 1</td></tr><tr><td>FEEDBACK1</td><td>{ KA V KW }</td><td>feedback type for pulse 1</td></tr><tr><td>ENG2</td><td></td><td>{ <i>weld_energy</i> }energy amount for pulse 2</td></tr><tr><td>FEEDBACK2</td><td>{ KA V KW }</td><td>feedback type for pulse 2</td></tr><tr><td>SQUEEZE</td><td>{ <i>squeeze_time</i> }</td><td>squeeze time</td></tr><tr><td>UP1</td><td></td><td>{ <i>weld_time</i> }up slope time of pulse 1</td></tr><tr><td>WELD1</td><td>{ <i>weld_time</i> }</td><td>weld time of pulse 1</td></tr><tr><td>DOWN1</td><td>{ <i>weld_time</i> }</td><td>down slope time of pulse 1</td></tr><tr><td>COOL</td><td></td><td>{ <i>weld_time</i> }cool time</td></tr><tr><td>UP2</td><td></td><td>{ <i>weld_time</i> }up slope time of pulse 2</td></tr><tr><td>WELD2</td><td>{ <i>weld_time</i> }</td><td>weld time of pulse 2</td></tr><tr><td>DOWN2</td><td>{ <i>weld_time</i> }</td><td>down slope time of pulse 2</td></tr><tr><td>HOLD</td><td></td><td>{ <i>hold_time</i> }hold time</td></tr></table>	ENG1	{ <i>weld_energy</i> }	energy amount for pulse 1	FEEDBACK1	{ KA V KW }	feedback type for pulse 1	ENG2		{ <i>weld_energy</i> }energy amount for pulse 2	FEEDBACK2	{ KA V KW }	feedback type for pulse 2	SQUEEZE	{ <i>squeeze_time</i> }	squeeze time	UP1		{ <i>weld_time</i> }up slope time of pulse 1	WELD1	{ <i>weld_time</i> }	weld time of pulse 1	DOWN1	{ <i>weld_time</i> }	down slope time of pulse 1	COOL		{ <i>weld_time</i> }cool time	UP2		{ <i>weld_time</i> }up slope time of pulse 2	WELD2	{ <i>weld_time</i> }	weld time of pulse 2	DOWN2	{ <i>weld_time</i> }	down slope time of pulse 2	HOLD		{ <i>hold_time</i> }hold time
ENG1	{ <i>weld_energy</i> }	energy amount for pulse 1																																						
FEEDBACK1	{ KA V KW }	feedback type for pulse 1																																						
ENG2		{ <i>weld_energy</i> }energy amount for pulse 2																																						
FEEDBACK2	{ KA V KW }	feedback type for pulse 2																																						
SQUEEZE	{ <i>squeeze_time</i> }	squeeze time																																						
UP1		{ <i>weld_time</i> }up slope time of pulse 1																																						
WELD1	{ <i>weld_time</i> }	weld time of pulse 1																																						
DOWN1	{ <i>weld_time</i> }	down slope time of pulse 1																																						
COOL		{ <i>weld_time</i> }cool time																																						
UP2		{ <i>weld_time</i> }up slope time of pulse 2																																						
WELD2	{ <i>weld_time</i> }	weld time of pulse 2																																						
DOWN2	{ <i>weld_time</i> }	down slope time of pulse 2																																						
HOLD		{ <i>hold_time</i> }hold time																																						

APPENDIX B: HOST ORIGINATED COMMANDS

squeeze_time and *hold_time* are the parameter that defines the time for the given period in 1 msec. Valid range is from 0 to 999.

weld_time is the parameter that defines the time for the given period in 0.1 msec. Valid range is from 0 to 990.

weld_energy is the parameter that specifies the amount of weld energy. In the current feedback mode, *weld_energy* is in unit of 0.001KA. In the voltage feedback mode, *weld_energy* is in units of 0.001V. In the power feedback mode, *weld_energy* is in units of 0.001KW.

command: **RELAY {READ | SET} <crLf>**
parameter_name value [<crLf>
parameter_name value <crLf>
] <lf>

HF25 State: Any except while welding

description: Provides control over the HF25 schedule parameters for relay settings. When used with the "**READ**" keyword, the relay settings of the currently loaded schedule are returned (see **RELAY** under HF25 ORIGINATED COMMANDS). When the "**SET**" keyword is used, the host may set (change) the value of one or more of the relay settings of the currently loaded schedule. The following is a list of valid literal substitutions for the *parameter_name* and *value* variables:

ACTIVE1 { **HIGH** | **LOW** } Relay 1 Active High or Active Low
CONDITION1 { **WELD** | **END** | **ALARM** | **OUT** } Relay 1 Active Conditions
ACTIVE2 { **HIGH** | **LOW** } Relay 2 Active High or Active Low
CONDITION2 { **WELD** | **END** | **ALARM** | **OUT** } Relay 2 Active Conditions
ACTIVE3 { **HIGH** | **LOW** } Relay 3 Active High or Active Low
CONDITION3 { **WELD** | **END** | **ALARM** | **OUT** } Relay 3 Active Conditions
ACTIVE4 { **HIGH** | **LOW** } Relay 4 Active High or Active Low
CONDITION4 { **WELD** | **END** | **ALARM** | **OUT** } Relay 4 Active Conditions

command: **MONITOR {READ | SET} <crLf>**
parameter_name value [<crLf>
parameter_name value <crLf>
]} <lf>

HF25 State: Any except while welding

description: Provides control over the basic weld monitor settings of the HF25 schedule. When used with the "**READ**" keyword, the basic weld monitor settings of the currently loaded schedule are returned (see **MONITOR** under HF25 ORIGINATED COMMANDS). When the "**SET**" keyword is used, the host may set (change) the value of one or more of the parameters of the basic weld monitor settings pertaining to the currently loaded schedule.

APPENDIX B: HOST ORIGINATED COMMANDS

The following is a list of valid literal substitutions for the *parameter_name* and *value* variables:

MONTYPE1 { **KA** | **V** | **KW** } Monitor Type for pulse 1
UPPER1 { *limit_value* } Upper Limit for pulse 1
LOWER1 { *limit_value* } Lower Limit for pulse 1
ACTION1 { **none** | **STOP** | **INHIBIT** } Out of Limit Action for pulse 1
MONTYPE2 { **KA** | **V** | **KW** } Monitor Type for pulse 2
UPPER2 { *limit_value* } Upper Limit for pulse 2
LOWER2 { *limit_value* } Lower Limit for pulse 2
ACTION2 { **none** | **STOP** | **INHIBIT** } Out of Limit Action for pulse 2

limit_value is the parameter that specifies the range of the valid readings. If the reading was within the range of the *limit_value*, no alarm will occur. If the reading was out of the valid range, an alarm will occur. If the monitor type is KA, the *limit_value* is in unit of 1A. If the monitor type is V, the *limit_value* is in unit of 1mV. If the monitor type is KW, the *limit_value* is in unit of 1W. The valid number for *limit_value* is 1 through 9999 and 0 is for none.

command: **SYSTEM** {**READ** | **SET** <crLf>
parameter_name value [<crLf>
parameter_name value <crLf>
]} <lf>

HF25 State: Any

description: Provides control over HF25 system parameters. When used with the "**READ**" keyword, all system parameters are returned (see **SYSTEM** under HF25 ORIGINATED COMMANDS). When used with the "**SET**" keyword, the host may set (change) the value of one or more of the system parameters. The following is a list of valid literal substitutions for the *parameter_name* and *value* variables:

LIGHT { *light_value* } LCD contrast
LOUDNESS { *loudness_value* } Buzzer Loudness
BUZZER { **OFF** | **ON** } end of cycle buzzer
DISPLAY { **PEAK** | **AVG** } Display mode
SWTYPE { **MECHANICAL** | **OPTO** | **PLC** } Input Switch Type
SWITCH { **OPEN** | **CLOSED** } Input Switch mode
CTTYPE { **MECHANICAL** | **OPTO** | **PLC** } Control Signals Type
CONTROL { **OPEN** | **CLOSED** } Control Signals mode
FIRESW { **AUTO** | **OPTO** | **NONE** } Firing Switch Type
GRAPH { **OFF** | **ON** } Update Graph
WELDABORT { **OFF** | **ON** } Footswitch weld abort
DEBOUNCE { **0** | **10** | **20** | **30** } Switch debounce time in msec

APPENDIX B: HOST ORIGINATED COMMANDS

These parameters pertain to the settings of the option menus available via the front panel user interface.

light_value is a number 0 to 99 for brightness of the LCD. 0 is dark and 100 is the brightest.

loudness_value is a number 0 to 99 for buzzer loudness. 0 is off and 100 is the loudest.

command: **SYNC** <crLf><lF>
HF25 State: Any
description: Provides synchronization of the commands. The HF25 return SYNC command back to the host computer.

Command: **CURRENT** <crLf><lF>
HF25 state: Any
Description: Requests the HF25 to report the sampled Current data of the last weld. HF25 shall return with **CURRENT** report. See CURRENT command under HF25 Originating Commands section.

Command: **VOLTAGE** <crLf><lF>
HF25 state: Any
Description: Requests the HF25 to report the sampled Current data of the last weld. HF25 shall return with **VOLTAGE** report. See VOLTAGE command under HF25 Originating Commands section.

Command: **POWER** <crLf><lF>
HF25 state: Any
Description: Requests the HF25 to report the sampled Current data of the last weld. HF25 shall return with **POWER** report. See POWER command under HF25 Originating Commands section.

APPENDIX C: HF25 ORIGINATED COMMANDS

These are the commands sent from a HF25 to a host computer.

- Command: **STATUS** *state_name* <crLf><lF>
HF25 state: Any
Description: Identifies the current status of the weld data buffer. May be in response with “**OK**” or “**OVERRUN.**” “**OK**” means that the HF25 weld buffer did not over-run since the last data collection and all the data are intact. “**OVERRUN**” means that the HF25 weld buffer did over-run since the last data collection and only the latest 3000 weld data are available to report.
- Command: **COUNT** *number* <crLf><lF>
HF25 state: Any
Description: Returns the number of weld data available in HF25. The total number of weld data that the HF25 holds in the buffer is 3000.
- Command: **SCHEDULE** *schedule_number* <crLf><lF>
HF25 state: Any
Description: Returns the current schedule number to the host. *schedule_number* may be any number from **0** to **127**.
- Command: **REPORT** *number_of_reports* <crLf>
 report <crLf> *report* <crLf>
 report <crLf><lF>
HF25 state: Any
Description: Returns the requested number of weld reports. First field is the number of reports to be sent. Then follows the packets of report. One report pack hold all the information about a weld. Each report packet is separated by <crLf> and this command ends with <crLf><lF>.
- number_of_reports*: This is the number of reports that shall be included in this command. If the host computer requests more weld data than is available in the weld data buffer, the HF25 sends only the weld reports in the weld buffer and the *number_of_reports* is the number of weld reports available in the weld data buffer. After the report is sent to the host computer, the HF25 erases the weld data sent to the host from the weld data buffer.
- report*: { *schedule_number* , *current_1* , *voltage_1* , *current_2* , *voltage_2* ,
 %_control_1, pulse_width, %_control_2, weld_status }

APPENDIX C: HF25 ORIGINATED COMMANDS

The fields in the report packet are separated with a comma and all fields are in integer format. There are always 8 fields in a report packet.

schedule_number: The schedule number of the weld
current_1: The average current of pulse 1 (in A)
voltage_1: The average voltage of pulse 1(in mV)
%_control_1: The % capacity control needed to reach the pulse 1
current_2: The average current of pulse 2 (in A)
voltage_2: The average voltage of pulse 2 (in mV)
%_control_2: The % capacity control needed to reach the pulse 2
weld_status: The status of the weld.

Command: **STATE** *state_name*<crLf><lF>
HF25 State: Any
Description: Identifies the current state of operation of the sentry. May be in response to the **STATE READ** command sent by the host, or may be sent as a result of a state change from the sentry front panel. *state_name* may be any of "RUN "PROGRAM or "MENU".

Command: **SCREEN** { **TEXT** | **ATTRIBUTE** } <crLf>
screen_dump<crLf><lF>
HF25 State: Any state
Description: Returns the request screen dump from the HF25. If TEXT is specified, the HF25 is sending 320 bytes of the text screen dump. If ATTRIBUTE is specified, the HF25 is sending 320 bytes of attribute screen dump.

Command: **COUNTER** *number* <crLf><lF>
HF25 state: Any
Description: Returns the requested current HF25 weld counter number.

Command: **SCHEDULE***schedule_number* <crLf>
FEEDBACK1{ **KA** | **V** | **KW** } <crLf>
FEEDBACK2{ **KA** | **V** | **KW** } <crLf>
SQUEEZE*squeeze_time* <crLf>
UP1*weld_time* <crLf>
WELD1*weld_time* <crLf>
DOWN1*weld_time* <crLf>
COOL*weld_time* <crLf>
UP2*weld_time* <crLf>
WELD2*weld_time* <crLf>
DOWN2*weld_time* <crLf>
HOLD*hold_time* <crLf>
ENG1*weld_energy* <crLf>
ENG2*weld_energy* <crLf>

APPENDIX C: HF25 ORIGINATED COMMANDS

- HF25 State: Any
Description: Reports the settings of the currently loaded HF25 schedule parameters. The *schedule_number* variable identifies which schedule is currently loaded, and may be any value from 0 to 99.
- squeeze_time* and *hold_time* are the parameter that defines the time for the given period in 1 msec. Valid range is from 0 to 999.
- weld_time* is the parameter that defines the time for the given period in 0.1 msec. Valid range is from 0 to 990.
- weld_energy* is the parameter that specifies the amount of weld energy. In the current feedback mode, *weld_energy* is in unit of 0.001KA. In the voltage feedback mode, *weld_energy* is in units of 0.001V. In the power feedback mode, *weld_energy* is in units of 0.001KW.
- Command: **RELAY** <crLf>
ACTIVE1 { **HIGH** | **LOW** } <crLf>
CONDITION1{ **WELD** | **END** | **ALARM** | **OUT** } <crLf>
ACTIVE2 { **HIGH** | **LOW** } <crLf>
CONDITION2{ **WELD** | **END** | **ALARM** | **OUT** } <crLf>
ACTIVE3 { **HIGH** | **LOW** } <crLf>
CONDITION3{ **WELD** | **END** | **ALARM** | **OUT** } <crLf>
ACTIVE4 { **HIGH** | **LOW** } <crLf>
CONDITION4{ **WELD** | **END** | **ALARM** | **OUT** } <crLf>
- HF25 State: Any
Description: Reports the relay settings.
- Command: **MONITOR** *schedule_number*<crLf>
MONTYPE1{ **KA** | **V** | **KW** }<crLf>
UPPER1{ *limit_value* }<crLf>
LOWER1{ *limit_value* }<crLf>
ACTION1{ **none** | **STOP** | **INHIBIT** }<crLf>
MONTYPE2{ **KA** | **V** | **KW** }<crLf>
UPPER2{ *limit_value* }<crLf>
LOWER2{ *limit_value* }<crLf>
ACTION2{ **none** | **STOP** | **INHIBIT** }<crLf>
<lF>
- HF25 State: Any
Description: Reports the settings of the basic weld monitor of the currently loaded HF25 schedule. The *schedule_number* variable identifies which schedule is currently loaded, and may be any value from 000 to 127. The possible value for all variables listed after their parameter name correspond to the values listed under **MONITOR** the *Appendix B, Host Originated Commands*.

APPENDIX C: HF25 ORIGINATED COMMANDS

- Command: **CURRENT** *number_of_data* <crLf>
 data <crLf> *data* <crLf> *data* <crLf><lf>
- HF25 state: Any
- Description:Returns the Current reading of the last weld. First field is the number of data to be sent. Then follows the packets of data. Each data is separated by <crLf> and this command ends with <crLf><lf>.
- number_of_data:* This is the number of data that shall be included in this command. The HF25 Control samples current every 50 μ s. For a weld less than 100 ms weld time, the number of data should be *total weld time* \div 50 μ s. This number will be always less than 2000.
- data:*an integer number in unit of A.
-
- Command: **VOLTAGE** *number_of_data* <crLf>
 data <crLf> *data* <crLf> *data* <crLf><lf>
- HF25 state: Any
- Description:Returns the Voltage reading of the last weld. First field is the number of data to be sent. Then follows the packets of data. Each data is separated by <crLf> and this command ends with <crLf><lf>.
- number_of_data:* This is the number of data that shall be included in this command. The HF25 Control samples Voltage every 50 μ s. For a weld less than 100 ms weld time, the number of data should be *total weld time* \div 50 μ s. This number will be always less than 2000.
- data:*an integer number in unit of mV.
-
- Command: **POWER** *number_of_data* <crLf>
 data <crLf> *data* <crLf> *data* <crLf><lf>
- HF25 state: Any
- Description:Returns the Power reading of the last weld. First field is the number of data to be sent. Then follows the packets of data. Each data is separated by <crLf> and this command ends with <crLf><lf>.
- number_of_data:*This is the number of data that shall be included in this command. The HF25 Control samples Current and Voltage every 50 μ s. For a weld less than 100 ms weld time, the number of data should be *total weld time* \div 50 μ s. This number will be always less than 2000.
- data:*an integer number in unit of W.

